(12) UK Patent Application (19) GB (11) 2 367 986 (13) A

(43) Date of A Publication 17.04.2002

(21) Application No 0106559.8

(22) Date of Filing 16.03.2001

(51) INT CL[7]
H04L 9/32

(52) UK CL (Edition T)
H4P PDCSA

(56) Documents Cited
GB 2313272 A US 6009528 A

(58) Field of Search
UK CL (Edition S) H4P PDCSA PPEB
INT CL[7] H04L 9/32 29/06
ONLINE: WPI, EPODOC, JAPIO.

(71) Applicant(s)
Telefonaktiebolaget L M Ericsson (publ)
(Incorporated in Sweden)
SE-126-25, Stockholm, Sweden

(72) Inventor(s)
Pekka Nikander

(74) Agent and/or Address for Service
Marks & Clerk
4220 Nash Court, Oxford Business Park South,
OXFORD, OX4 2RU, United Kingdom

(54) Abstract Title
IP network authorisation using coded interface identifier part of IP address

(57) A method of verifying that a host coupled to an IP network is authorised to use an IP address which the host claims to own, the IP address comprising a routing prefix and an interface identifier part. The method comprises receiving from the host one or more components, applying a one-way coding function to the or each component and/or derivatives of the or each component, and comparing the result or a derivative of the result against the interface identifier part of the IP address. If the result or its derivative matches the interface identifier the host is assumed to be authorised to use the IP address and if the result or its derivative does not match the interface identifier the host is assumed not to be authorised to use the IP address. A method for authenticating a public key and a method for generating the interface identifier part of an IP address are also disclosed.

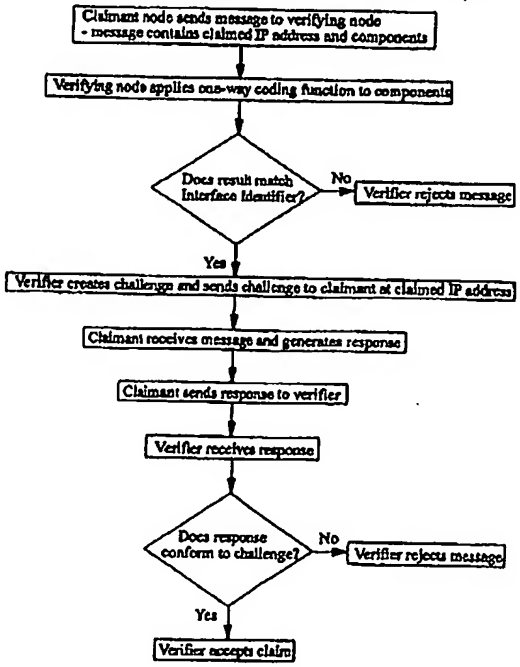

Figure 3

At least one drawing originally filed was informal and the print reproduced here is taken from a later filed formal copy.

This print takes account of replacement documents submitted after the date of filing to enable the application to comply

GB 2 367 986 A

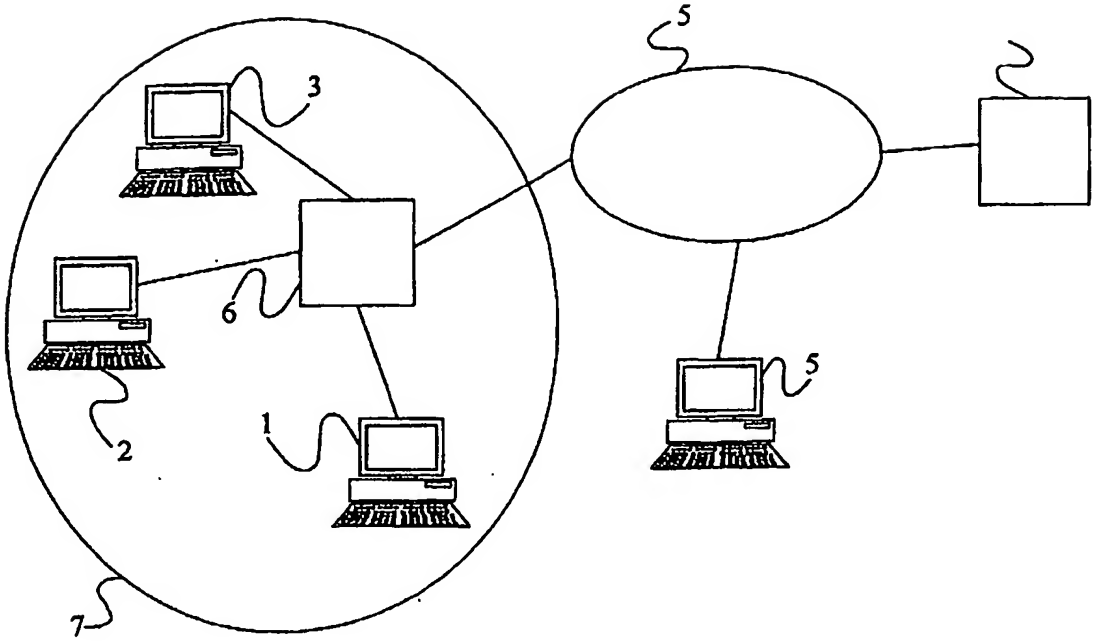

Figure 1

ADDRESS MECHANISMS IN INTERNET PROTOCOL

Field of the Invention

The present invention relates to address mechanisms in Internet Protocol (IP) and more particularly to address mechanisms in IPv6.

Background to the invention

The massive growth in the use of the Internet has exposed weaknesses and limitations of the current Internet protocol known as IPv4. The Internet Engineering Task Force (IETF), which is a loose connection of interested parties, is therefore developing an enhanced Internet protocol known as IPv6. In particular, IPv6 incorporates much improved security mechanisms known as IPSec which enable two or more parties to communicate securely over the Internet, as well as provisions for mobile Internet access (mobileIP). MobileIP allows users to access the Internet on the move, roaming from one IP access node to another. MobileIP will be used in particular by those accessing the Internet via wireless mobile devices (connected for example to wireless LANs and cellular telephone networks).

IPv6 provides for a much larger IP address space, providing IP addresses of 128 bits in length. The first 64 bits of an address form a routing prefix which uniquely identifies the Internet access node (or so-called "local link") used by an IP terminal or host, whilst the last 64 bits form a host suffix which uniquely identifies the mobile terminal to the access node (or within the local link). The host suffix is referred to as an "interface identifier" as it identifies the host uniquely over the access interface. Typically, when a host registers with an access node, the host learns the routing prefix of the access node from an advertisement message sent from the access node. According to RFC3041 (IETF), a host then generates its interface identifier using a random number generated by the host. The host may additionally use a link layer address to generate the interface identifier, the link layer address being for example a MAC layer address used by the access network.

A potential problem with this approach is that two hosts connected to the same access node (and therefore having the same routing prefix) may generate the same interface identifiers and therefore the same IP addresses. This cannot be allowed. IPv6 therefore provides a mechanism known as Duplicate Address Detection (DAD). Once a host has generated a potential IP address, it sends a neighbour solicitation message containing the proposed IP address to the access node or directly to the local link if the local link provides for local broadcast or multicast. If required, the access node broadcasts the message to all other hosts connected to the node. If a host receiving the message recognises the IP address contained in the message as an address which it has already adopted, that hosts responds by sending to the soliciting host a neighbour advertisement. If a soliciting host does not receive a neighbour advertisement message within some predefined time, it will adopt the generated address. If on the other hand a neighbour advertisement is received within the predefined time, the soliciting host will generate a new interface identifier and IP address, and repeat the solicitation process.

A problem with the approach described above is that it can be relatively simple for malicious third party to deny the soliciting node access by always responding to a neighbour solicitation message with a neighbour advertisement message. This kind of attack is known as a "denial of service" attack.

A further problem can arise in IPv6 with mobileIP. As already mentioned, mobileIP allows hosts to roam between access nodes and even access networks, a feature which requires that hosts be allowed to change the IP addresses which define their current physical locations. Typically, a mobile host is allocated a "fixed" home IP address in a home network. When the host is at home, it can use this home address as its physical address. However, when a host attaches itself to a "foreign" access node, the host is allocated a temporary care-of-address. Hosts corresponding with the mobile host maintain a binding cache containing mappings between home addresses and care-of-addresses. For incoming packets, the mobileIP layer at a correspondent host exchanges the care-of-address for the home address in the destination field, whilst for outgoing packets the mobile IP layer at a correspondent host exchanges the home address for the care-of-address in the destination address field. When a mobile host obtains a new care-

of-address, it must send a binding update message to all correspondent hosts in order to update their binding caches.

A potential risk of this mechanism is that a malicious third party may be able to send a falsified binding update to a correspondent host to cause data packets intended for a mobile host to be routed to the malicious party. If the packets are then forwarded to the mobile host by the malicious party (after being opened and read by that party), the mobile host may not even know that its packets are being re-routed and read. This problem is not limited to mobileIP, but is also present in other signalling functions within the IPv6 architecture. The mobileIP related problem and some of the other problems are described in more detail in the IETF submission "draft-nikander-ipng-address-ownership-00.txt" of February 2001.

A solution to this problem has been proposed in the IETF submission "draft-bradner-pbk-frame-00.txt" of February 2001. This involves generating at the mobile host a purpose built key (PBK) pair comprising a public and a private key. An Endpoint ID (EID) is generated at the mobile host by applying a hash to the public key. The EID is sent to a correspondent host soon after initiation of an IP connection. Subsequently, the mobile host sends to the correspondent host the public key – the correspondent host is able to verify that the public key "belongs" to the connection by applying the one-way coding function to the key and comparing the result with the previously received EID. Any binding update which is subsequently sent, is signed at the mobile host with the host's private key. The correspondent host verifies the signature attached to the binding update using the previously received public key.

Summary of the Present Invention.

Two problems with IPv6 have been considered above, namely the possibility of a denial of service attack being launched by the sending of neighbour advertisement messages to a soliciting host, and a "man-in-the-middle" attack based upon falsification of binding updates. Similar problems might arise in the following situations; ICMP Router discovery (RFC2461 Section 6.1.2), ICMP Redirect (RFC2461 Section 8.1), Generic Tunnelling (RFC2473), IPsec Tunnelling (RFC2401), Router Renumbering (RFC2894),

IPv6 Routing Header (RFC2460 Section 8.4), and possibly in the Inverse Neighbour Discovery (draft-ietf-ion-ipv6-ind-05.txt) and SCTP (RFC2960). It may also arise in the HIP proposal, as well as a number of other proposals. All of these problems have a common cause - it is not possible to verify the ownership of an IP address.

Considering the proposal by Bradner, the PBK does not bind the public key to an IP address but rather only to the EID. Furthermore, there is no direct binding between the EID and the IP address. Thus, the PBK does not directly remedy the problems described.

It is an object of the present invention to overcome the above noted problems. In particular it is an object of the present invention to provide a means for proving ownership of an IP address. In this document, ownership of an IP address denotes that the owner is authorised to use the IP address within the specified scope of the address and is authorised to change routing information that applies to the IP address.

According to a first aspect of the present invention there is provided a method of verifying that a host coupled to an IP network is authorised to use an IP address which the host claims to own, the IP address comprising a routing prefix and an interface identifier part, the method comprising receiving from the host one or more components, applying a one-way coding function to the or each component and/or derivatives of the or each component, and comparing the result or a derivative of the result against the interface identifier part of the IP address, wherein if the result or its derivative matches the interface identifier the host is assumed to be authorised to use the IP address and if the result or its derivative does not match the interface identifier the host is assumed not to be authorised to use the IP address.

It will be appreciated that a host will generate the interface identifier part of its IP address using the component(s) and/or derivatives of the component(s). Where a plurality of components are involved, the one-way coding function may be applied to a combination of certain of the components and derivatives of others of the components. The generating host will retain these components during a connection and will be able to provide them to some other party when required. That other party can use the

components to reconstruct the interface identifier and verify the ownership of the IP address by the host. It is difficult for a malicious third party to reverse the coding and recover the components from the IP address, and therefore to impersonate the true owner of an address.

Said one-way coding function may be SHA-1, MD5, or any other cryptographically known one-way coding function.

An advantage of embodiments of the present invention is that they do not require any global infrastructure, such as Public Key Infrastructure (PKI), but are based on a novel application of cryptographic functions. Furthermore, since certain embodiments of this invention do not require any architectural changes to the currently proposed IPv6 specifications, the present invention is more advantageous than the Bradner proposal considered above, which would require changes in the currently proposed IPv6 architecture.

The IP network may comprise the Internet, or a private IP network such as a corporate LAN or WAN. The IP network may comprise an access network coupled to the Internet and/or a private IP network.

Preferably, said components comprise a public key or a digest of a public key generated by said host or issued to said host by some other party, or a fixed (e.g. zero) bit sequence of the same length, and a hash value being one of a sequence of related hash values. Alternatively or in addition, said components comprise an initial interface identifier which corresponds to or is derived from a link layer address of the host, or a fixed (e.g. zero) bit sequence of the same length. More preferably, said components comprise both said public key or said digest of a public key and said initial interface identifier which corresponds to or is derived from a link layer address of the host, or a fixed (e.g. zero) bit sequence of the same length. More preferably, said components comprise a counter value which identifies the position of the received hash value in said sequence.

Preferably, said series of hash values are derived at the host by applying a one-way coding function to a seed value, said public key or digest of the public key, and said initial interface identifier. Alternatively, said series of hash values are derived at the host by applying a one-way coding function to said seed value and either said public key or digest of the public key, or said initial interface identifier. The hash value which is derivable from the received hash value, and which is used to generate said result, is the last hash value in the sequence. In the event of a first IP address verification, the hash value received from the host is the hash value preceding the final hash value in the sequence. For each subsequent verification process, the next previous hash value must be received.

Preferably, the method comprises deriving the final value of the hash sequence and applying a one-way coding function to that final value concatenated with one or more other components. The result may be further processed, before comparing the final result with the interface identifier.

According to a second aspect of the present invention there is provided a method of generating an IP address at a host, the IP address comprising a routing prefix and an interface identifier part, the method comprising generating the interface identifier part by applying a one-way coding function to one or more components.

Preferably, said components include a hash value which is generated through some method that uses information from a random number. More preferably, the hash value is generated by applying a one-way coding function to a combination of the random number and an initial interface identifier, a public key or a digest of the public key, or a combination of said initial interface identifier and a public key or a digest of the public key.

According to a third aspect of the present invention there is provided a method of avoiding the duplication of IP addresses in an IP network when a new host attaches to the network, the method comprising the steps of:

generating an Interface Identifier at the new host by combining a component or components and/or derivatives of the component or components using a one-way

coding function and using the result or a derivative of the result as said interface identifier, said interface identifier forming part of said IP address;

sending a neighbour solicitation message from the new host to other hosts already attached to the access network;

receiving a neighbour advertisement message at the new host from each other host claiming to own said IP address, the or each neighbour advertisement message containing said component(s); and

for each received neighbour advertisement message

combining the component(s) and/or derivatives of the component(s) using said coding function; and

comparing the result or a derivative of the result against the interface identifier part of the IP address, wherein if the result or the derivative matches the interface identifier the host from which the neighbour advertisement message is received is assumed to be authorised to use the IP address and if the result or its derivative does not match the interface identifier that host is assumed not to be authorised to use the IP address.

Preferably, said component(s) include a public key, a digest of a public key, or a derivative thereof. This allows the new host to learn a public key such that the new host may safely assume that the host sending the advertisement has used the public key to generate its IP address, and therefore the new host may assume that in order to be authorised the IP address the said other host must posses the corresponding private key.

Preferably, based on the assumption noted above, the new host may use any well known authentication protocol or other public key cryptography based protocol (such as zero knowledge protocols) to verify that the another host does currently possess the necessary private key. Preferably, successful running of the (verification) protocol is assumed to denote that the other host is authorised to use the IP address, and failure to successfully running the said protocol is assumed to denote that the other host is not authorised to use the IP address.

According to a fourth aspect of the present invention there is provided a method of verifying that a host coupled to an IP network is authorised to use an IP address which

the host claims to own, and that the host is able to receive data packets sent to that address, the method comprising:

carrying out the method of the above first aspect of the invention to confirm that said host is authorised to use the IP address;

sending a challenge to the host using said IP address as the destination address for the challenge;

receiving a response from the host; and

verifying that the received response is a correct response to the challenge.

Preferably, said challenge comprises a randomly generated number. More preferably, said challenge comprises said IP address concatenated with a randomly generated number. More preferably, said challenge is constructed by applying a one-way coding function to said IP address concatenated with a randomly generated number.

Preferably, said response comprises the challenge. More preferably, said response comprises the IP address concatenated with the challenge. More preferably, said response is constructed by applying a one-way coding function to said IP address concatenated with said challenge.

According to a fifth aspect of the present invention there is provided a method of authenticating a public key transmitted over an IP network from a first to a second host, the method comprising:

at said first host, generating an interface identifier using said public key and combining the interface identifier with a routing prefix to form an IP address for the first host;

sending said public key from the first to the second node over said IP network; and

at said second node, verifying that said public key was the key used to generate said interface identifier.

According to a sixth aspect of the present invention there is provided a method of binding an IP address to a public key, the IP address comprising a routing prefix and an interface identifier part, the method comprising:

generating said interface identifier by combining one or more components and/or derivatives of the components using a coding function; and

generating a certificate, signed with a private key of a public-private key pair, the certificate containing the interface identifier and ones of said components or said derivatives or further derivatives, such that the certificate can be authenticated using the host's public key, and ownership of the interface identifier can be verified by reconstructing the interface identifier using the contents of the certificate, and comparing the result against the true interface identifier.

Brief Description of the Drawings

Figure 1 illustrates schematically a network comprising a number of hosts coupled to the Internet;

Figure 2 is a flow diagram showing a Duplicate Address Detection process in the network of Figure 1;

Figure 3 is a flow diagram illustrating a method of performing a binding update in the network of Figure 1.

Detailed Description of Certain Embodiments

There is illustrated in Figure 1 an Internet communication scenario in which a number of user terminals or hosts 1 to 4 are coupled to the Internet 5. Hosts 1 to 3 are coupled to the Internet 5 via an access node 6 of an access network 7. Host 4 is connected by an access network which is not shown in the Figure.

Assume that one of the hosts 1 is new to the access network 7, and for that host the access network is a foreign network (and hence the access node 6 is a foreign agent). The host 1 discovers this fact by receiving a Router Advertisement message from the foreign agent (this message may be a message broadcast periodically by the foreign agent, or may be sent to the host 1 in response to a Router Solicitation message sent to the foreign agent from the host 1). The host 1 learns from the Router Advertisement message a routing prefix which uniquely identifies the foreign agent within the Internet. The host 1 then sends a Binding Update message to the home agent 8 of its home

network 9, via the foreign agent 6, to inform the home agent of its new location. The home agent responds by sending a Binding Acknowledgement message to the host 1 via the foreign agent 6. As already explained above, the host 1 combines the routing prefix and an interface identifier to form an IP address.

A new method for generating interface identifiers will now be described. This method has several advantages including:

- binding of the interface identifier to the link layer address;
- binding of the interface identifier to a public key;
- it provides a means to block a Denial-of-Service attack during Duplicate Address Detection;
- it provides a means to prove "address ownership" over a distance.

General description of the interface identifier

The method described is based on a cryptographically strong one-way coding function. A one-way coding function is denoted with HASH(...), and the particular one-way coding function used here is SHA-1 (although others may alternatively be used). In very general terms, the proposed method for generating an interface identifier is:

interface identifier := HASH(component1 | component2 | component3)

where "... | ..." denotes concatenation, one of the components is a newly generated ("fresh") random number, and the other components are pieces of information that the node (or host) generating the interface identifier wants to bind the interface identifier to.

Given an interface identifier, it is computationally difficult to compute a set of components that hash to that interface identifier value. As the interface identifier is 64 bits long, 63 bits of which are significant in this context, on average it takes (2^63)/2 = 2^62 operations to find such a set of components. Whilst it may be possible to reverse a hash value of that length with present day equipment, in practice it is likely to take several hundreds of years. Since the random interface identifiers are assumed to have a relatively short lifetime, at most in the order of days, this poses a negligible risk. Even